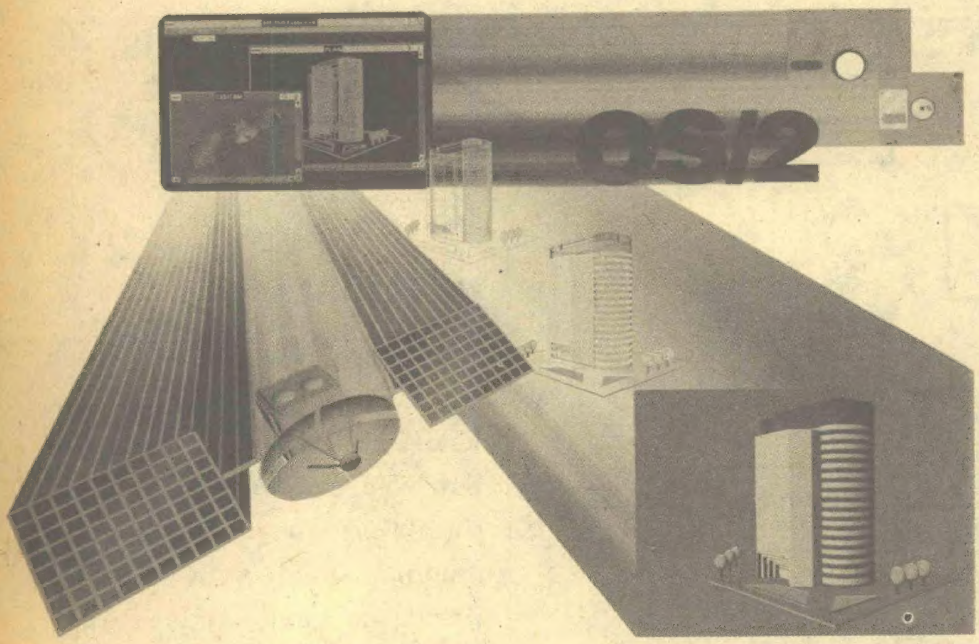
แนะนำ OS/2



กองบรรณาธิการ

บทความแนะนำ OS/2 ในฉบับนี้จะเป็นการกล่าวถึงเรื่องของตัว OS/2 และสถาปัตยกรรมในระดับผู้ใช้ คือยังไม่ลงลึกไปถึงโครงสร้างแกนหลักที่ซับซ้อนนัก ทั้งนี้ก็เพราะจุดประสงค์ที่จะให้ท่านผู้อ่านมองภาพของ OS/2 ได้กว้างๆ ในขณะเดียวกันก็ทราบถึงสมรรถนะ ขีดความสามารถ โดยเปรียบเทียบกับเอ็มเอสดอสที่เราใช้งานกันอยู่ โดยเฉพาะอย่างยิ่งบทความนี้ถือว่าเป็นการเปิดตัวเราเข้าไปรู้จักทักทายกับ OS/2 เนื่องจากอีกไม่นานต่อไปเราคงไม่อาจหลีกเลี่ยงการใช้เจ้า OS/2 นี้ไปได้

โอเอสทู...โอเอส 286?

ถ้าเราย้อนหลังไปตอนที่ไอบีเอ็มเอที ออกสู่ตลาดใหม่ๆ จนกระทั่งบัดนี้ก็เป็นเวลาถึง 3 ปีแล้ว ผู้ใช้เครื่องเอทีในตอนนั้นก็ไม่ได้มีข้อได้เปรียบกับผู้ใช้เครื่องเอกซ์ทีเท่าใด นอกจากความเร็วที่สูงขึ้นด้วยสมรรถนะของซีพียู 80286 แต่ในที่สุด Opereting System/2 ที่เราเรียกกันติดปากว่า OS/2 ก็เปิดตัวออกมาเป็นโปรแกรมจัดระบบงานใหม่ของเครื่องระดับเอที หรือเพื่อใช้ 80286 ให้เต็มขีดความสามารถ แต่อย่างไรก็ตาม 80286 ก็ไม่ได้เป็นซีพียูในระดับแนวหน้าสุดแล้ว แต่เป็น 80386 ซึ่งก็กำลังรอคอยโปรแกรมจัดระบบงานตัวใหม่อยู่ด้วยเช่นกัน

ในระดับผู้ใช้แล้ว จุดเด่นหลักของ OS/2 คือมองขนาดหน่วยความจำของเครื่องได้เกิน 1 เมกะไบต์ และมีระบบมัลติทาสกิ้งหรือทำงานหลายงานได้พร้อมๆกัน โอเอสทูสามารถมองหน่วยความจำของ 80286 ได้เต็มที่คือขนาด 16 เมกะไบต์ รวมทั้งมีระบบจัดการหน่วยความจำเสมือน (Virtual Memory) ซึ่งทำให้เครื่องใช้หน่วยความจำเกินกว่าที่มีอยู่ในหน่วยความจำหลักได้ ระบบหน่วยความจำเสมือนนี้ขยายได้ 1 กิกะไบต์

ต้องมีอะไร ถ้าจะใช้โอเอสทู

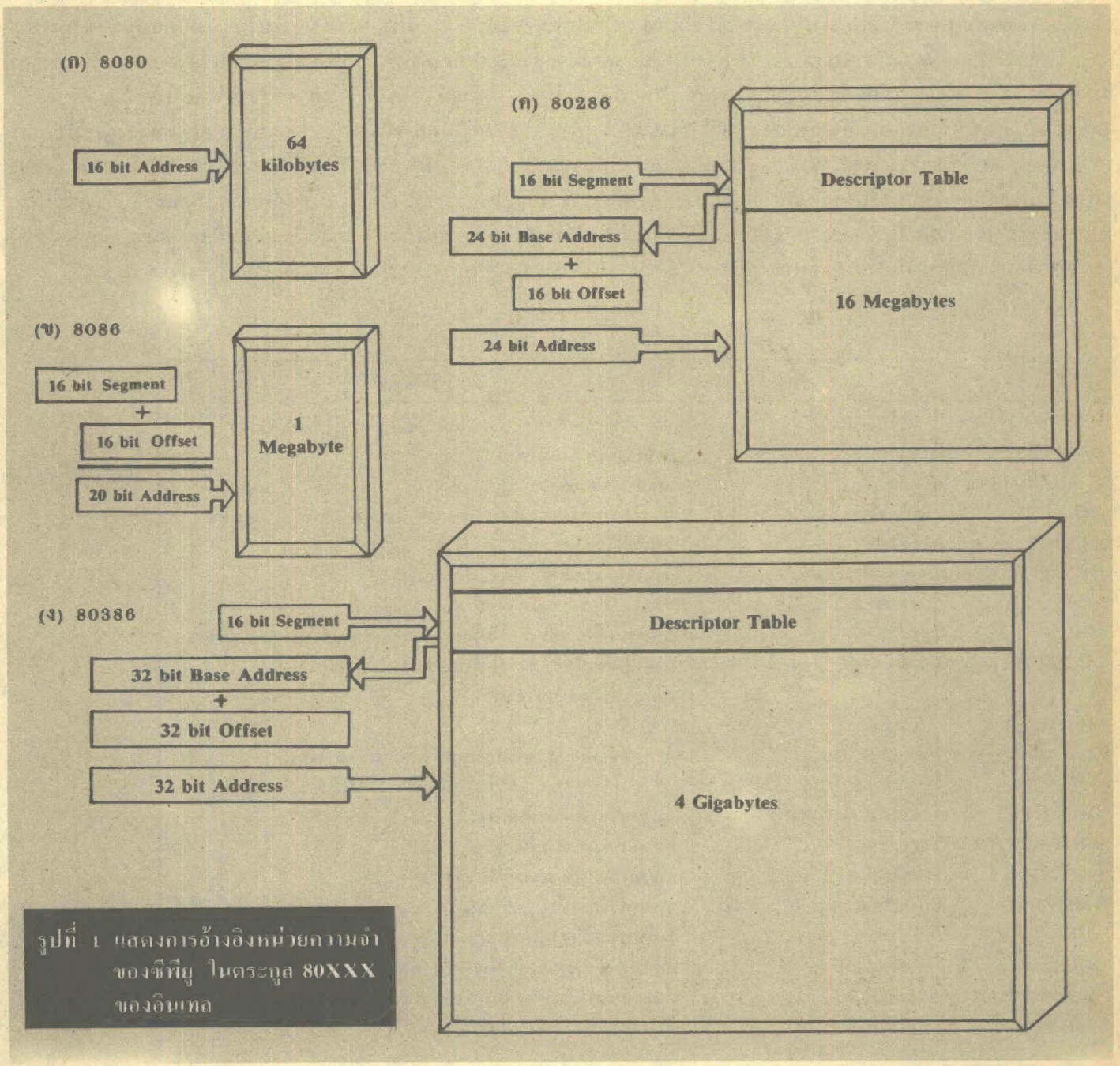
การใช้ OS/2 จะใช้กับเครื่อง 80286 หรือ 80386 ซึ่งมีฮาร์ดดิสก์และหน่วยความจำหลักขนาด 1 เมกะไบต์ อย่างไรก็ตามหน่วยความจำเพียง 1 เมกะไบต์นี้ก็ยังไม่สามารถจะใช้กับโปรแกรมประยุกต์ของเอ็มเอสดอสได้ หน่วยความจำขนาดต่ำสุดจึงต้องเป็น 1.5 เมกะไบต์ หรือ 2 เมกะไบต์ เพื่อการใช้งานให้ได้เต็มที่ขึ้น การใช้ OS/2 ต้องมีอะแดปเตอร์ของจอสี ในขณะนี้อะแดปเตอร์ที่ใช้ได้มีอยู่ 2 ชนิดคือ Enhanced Graphics Adapter (EGA) และ Video Graphics Array (VGA) เท่านั้น

เราจึงกล่าวได้ว่า OS/2 จะใช้ได้สำหรับเครื่องระดับ AT ขึ้นไป และ PS/2 รุ่น 50, 60 และ 80 ยกเว้นรุ่น 30 จะใช้ OS/2 ไม่ได้เนื่องจากเครื่องรุ่นนี้ใช้ซีพียู 8086 แนวการออกแบบ OS/2 จะต่างกับโปรแกรมจัดระบบงาน UNIX ซึ่งตัว UNIX ออกแบบให้ยืดหยุ่นตัวกับระบบฮาร์ดแวร์ของเครื่องอื่นๆที่ต่างสถาปัตยกรรมกัน ในขณะที่ OS/2 นั้นพยายามจะปรับตัวให้เข้ากับสถาปัตยกรรมตามแบบอินเทล (คือ ซีพียูกลุ่ม 80XXX) ให้มากที่สุด ตัว OS/2 จึงไม่สามารถถ่ายเทตัวเองหรือปรับปรุงไปใช้กับเครื่องที่มีสถาปัตยกรรมต่าง

แบบได้

ขอให้เราหันกลับมามองซีพียู 80286 และ 80386 ซีพียูทั้ง 2 ตัวนี้มีความสามารถทำงานทั้งใน Real Mode และ Protected Mode ในสภาวะที่เรียกว่า Real Mode ตัวซีพียูจะจำลองตัวเหมือนกับซีพียู 8086 ที่มีความเร็วสูง และมีหน่วยความจำจำกัดได้เพียง 1 เมกะไบต์ ส่วนในสภาวะ Protected Mode เป็นสภาวะการทำงานเต็มตัวของซีพียูแต่ละตัวคือ 80286 มองเห็นหน่วยความจำขนาด 16 เมกะไบต์ และระบบความจำเสมือนขนาด 1 กิกะไบต์ ในขณะที่ 80386 จะมองหน่วยความจำหลักได้ถึง 4 กิกะไบต์ และมีระบบความจำเสมือนขยายไปได้ถึง 64 ทริลเลียนไบต์ (ทริลเลียน หรือ TB = จำนวนล้าน ล้าน หนคือเลข 13 ตำแหน่ง) ขอให้ดูรูปแสดงการมองหน่วยความจำของซีพียูในตระกูล 80XXX ดังรูปที่ 1 หลายคนอาจจะยังไม่เข้าใจถึงความหมายของคำว่า Protected Mode ซึ่งแท้ที่จริงแล้วคำนี้ใช้ในความหมายของการป้องกันหน่วยความจำแต่ละส่วนที่ถูกแบ่งใช้สำหรับงานมัลติทาสกิ้ง ซึ่งเป็นวิธีเดียวในการรักษาสภาพระบบรวมให้ทำงานในสภาวะมัลติทาสกิ้งได้

วิธีการป้องกันที่ว่านี้อาจทำได้โดยจัดสรรหน่วยความจำเป็นส่วนๆ สำหรับงานแต่ละงาน หน่วยความจำแต่ละส่วน



รูปที่ 1 แสดงการอ้างอิงหน่วยความจำของซีพียู ในตระกูล 80XXX ของอินเทล

ก็จะแยกออกจากกันไม่เกี่ยวข้องกัน ตัวฮาร์ดแวร์จะเป็นผู้คอยตรวจสอบว่ามีการใช้หน่วยความจำข้ามส่วนไปหรือไม่ ซึ่งถ้าเกิดปัญหานี้ขึ้นมาก็จะมีการขัดจังหวะหยุดการทำงานนั้นแล้วส่งการทำงานกลับไปยังโปรแกรมจัดระบบงาน วิธีการนี้ดีตรงที่ว่าฮาร์ดแวร์จัดการเองหมด โดยไม่ต้องพึ่งซอฟต์แวร์เลย

แต่ในกรณีของโอเอสทูนั้นใช้อีกวิธีหนึ่ง กล่าวคือจัดฟังก์ชันของการทำงานออกเป็นชั้นๆที่เรียกว่า Privilege Level ซึ่งมีทั้งหมด 4 ระดับคือ ระดับ 0 ถึง 3 แต่ละระดับจะมีความสำคัญไม่เท่ากัน คือ ระดับ 0 จะมีความสำคัญสูงสุด และระดับ 3 มีลำดับความสำคัญต่ำสุด ระดับที่สำคัญกว่าจะสามารถทำคำสั่งฮาร์ดแวร์คำสั่งใดๆก็ได้ และลดหลั่นกันไปตามลำดับ เรื่องของ Privilege level จะขอพักไว้ตรงนี้ก่อน เนื่องจากวิธีการจะเกี่ยวพันกับฮาร์ดแวร์ของซีพียูโดยตรงอีกด้วย

ดูการทำงานเริ่มต้นของ OS/2

การเริ่มต้นระบบของ OS/2 ภายหลังจากการโหลดตัวเองเข้าสู่หน่วยความจำแล้วก็มีการทำงานเริ่มต้นคล้ายคลึงกับดอส คือการอ่านไฟล์ CONFIG.SYS เพื่อปรับระบบ คำสั่งปรับระบบที่ใช้ได้กับ OS/2 แสดงไว้ดังตารางที่ 1 ส่วนใหญ่แล้วคำสั่งจะยังคงคล้ายคลึงกับดอส สำหรับคำสั่งใหม่ๆที่เพิ่มเข้ามาก็จะมีความสามารถในการทำงานในสภาพมัลติทาสกิ้งและระบบความจำเสมือน ถ้าสังเกตให้ดีเราจะพบว่าในตารางที่ 1 ไม่มีคำสั่งกำหนดจำนวนไฟล์ใช้งานสูงสุด (คำสั่ง FILE) เนื่องจาก OS/2 จะจัดการกับไฟล์แบบไดนามิก คือไม่จำกัดจำนวนไฟล์ใช้งาน

ภายหลังการปรับระบบแล้ว OS/2 จะเข้าสู่ Protected Mode และมองหาคอมมานด์โปรเซสเซอร์ หรือ CMD.EXE และรันแบตช์ไฟล์ STARTUP.CMD โดยอัตโนมัติ (แบตช์ไฟล์ใน Protected Mode จะมีนามสกุล .CMD เพื่อแยกความแตกต่างกับแบตช์ไฟล์ใน Real Mode ที่มีนามสกุล .BAT) ในกรณีที่แบตช์ไฟล์ STARTUP ไม่ได้รันโปรแกรมใดๆ ตัวคอมมานด์โปรเซสเซอร์ก็จะส่งเครื่องหมายเตรียมพร้อมรอการสั่งงานจากผู้ใช้

การเชื่อมต่อระหว่างผู้ใช้กับคอมมานด์ไลน์ในโหมดปกติของโอเอสจะเหมือนกับเอ็มเอสดอส ไม่ว่าจะเป็นการทำไปป์ การเปลี่ยนทิศทางอินพุตเอาต์พุต หรือการเรียกใช้โปรแกรม

คำสั่ง	พารามิเตอร์	การใช้งาน
BREAK	ON/OFF	ปรับการตรวจสอบ CTRL-BREAK เฉพาะ Real Mode
BUFFERS	ตัวเลข	กำหนดจำนวนบัฟเฟอร์
CODEPAGE	รหัส	เลือก code page
COUNTRY	รหัส	เลือกระบบการแสดงผล วัน เวลา การเงิน ตามรหัสประเทศ
DEVICE	ชื่อไฟล์	ติดตั้งดีไวซ์ไดรเวอร์
DEVINFO	อุปกรณ์, ชื่อไฟล์	เตรียมอุปกรณ์เพื่อรับตาราง code page
FCBS	ตัวเลข	จำนวน FCB ที่เปิดได้ใน Real Mode
IOPL	YES/NO	อนุญาตหรือไม่อนุญาตให้มีการเข้าถึงฮาร์ดแวร์ ถ้าโปรแกรมขอมา
LIBPATH	ชื่อเส้นทาง	กำหนดตำแหน่งของคลังโปรแกรมสำหรับการโหลดเพื่อนำมาลิงก์แบบไดนามิก
MAXWAIT	วินาที	
MEMMAN	[NO] SWAP, [NO] MOVE	กำหนดข้อเลือกสำหรับระบบหน่วยความจำเสมือน
PRIORITY	ABSOLUTE/DYNAMIC	กำหนดข้อเลือกการจัดลำดับงาน
PROTECTONLY	YES/NO	อีนาเบิ้ลการทำงานใน Real Mode
PROTSHELL	ชื่อไฟล์	คอมมานดโปรเซสเซอร์ในโปรเทกโหมด
RMSIZE	กิโลไบต์	ขนาดพาร์ติชันใน Real Mode
RUN	ชื่อไฟล์	รันโปรเซสเมื่อเริ่มต้นระบบ
SHELL	ชื่อไฟล์	คอมมานด์โปรเซสเซอร์ใน Real Mode
SWAPPATH	เส้นทาง	ตำแหน่งไฟล์ในการสลับหน่วยความจำ
TIMESLICE	มิลลิวินาที	ค่า timeslice มากสุดและน้อยสุด

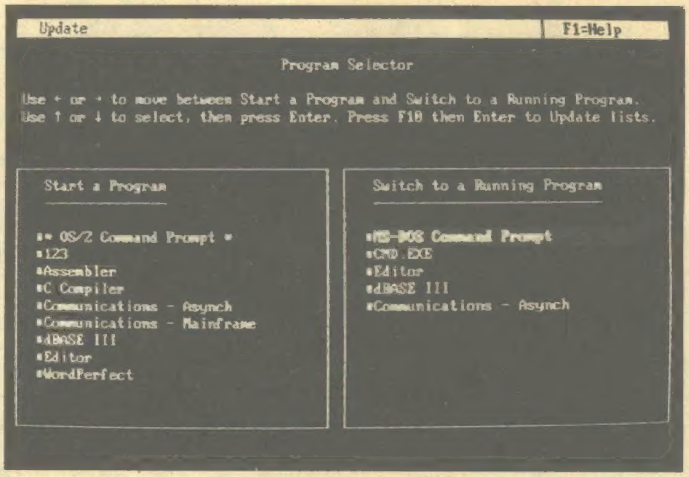
ตารางที่ 1 คำสั่งปรับระบบ

ส่วนในสภาวะ Protected Mode การสั่งงานจากเครื่องหมายเตรียมพร้อมจะมีลูกเล่นมากขึ้น เช่น พิมพ์คำสั่งหลายๆคำสั่งในบรรทัดเดียวกันได้ (ใช้เครื่องหมาย "&" คั่นคำสั่ง) หรือใช้การตรวจสอบทางลอจิคัลเพื่อเลือกหรือตัดสินใจการทำงาน ตัวอย่างเช่น ใช้เครื่องหมายลอจิคัล AND ซึ่งแทนด้วยสัญลักษณ์ "&&" ตัวอย่างเช่น คำสั่งต่อไปนี้จะเป็นการลบไฟล์ TEST.BAK ถ้ามีไฟล์ TEST.TXT อยู่

DIR TEST.TXT && DEL TEST.BAK

คำสั่งนี้หมายถึง ถ้าการค้นพบ TEST.TXT เป็นจริง คือมีไฟล์ TEST.TXT อยู่ คำสั่งที่อยู่หลัง "&&" จะถูกทำงาน ซึ่งคือคำสั่งลบไฟล์ TEST.BAK นั่นเอง

การใช้โปรแกรมในสภาพมัลติทาสกิ้ง และการสลับไปมาระหว่างโฟร์กราวนด์กับแบ็คกราวนด์จะถูกควบคุมโดยส่วนของ OS/2 ที่เรียกว่า "Session Manager" รูปแบบการทำงานของ Session Manager เป็นเมนูแบบฟูลสกรีน ตัว Session Manager สามารถเรียกออกมาใช้งานได้ตลอดเวลา ถึงแม้ว่ากำลังจะรันโปรแกรมใดอยู่ โดยการกดปุ่ม Ctrl-Esc ซึ่งมีผลทำให้โปรแกรมที่ทำงานอยู่ถูกปรับให้สู่แบ็คกราวนด์ และตัว Session Manager จะเริ่มทำงาน ดังรูปที่ 2 ซึ่งจะเห็นจอภาพถูกแบ่งออกเป็น 2 ส่วนคือ ทางซ้ายแสดงรายชื่อโปรแกรมที่เลือกมาทำงานได้ ส่วนทางขวาจะแสดงโปรแกรมที่เลือกทำงานไว้แล้ว การทำงาน Session Manager ทั้งหมดจะเป็นแบบฟูลสกรีนที่ใช้เคอร์เซอร์คีย์ควบคุมทั้งหมด



รูปที่ 2
Session Manager

แง้มดูโครงสร้าง OS/2

ตัว OS/2 มีโครงสร้างที่ต่างไปจากดอสมาก ตัวโอเอสทูมีโครงสร้างดังรูปที่ 3 คือมีโครงสร้างสำคัญ 3 ส่วน ซึ่งถูกประกบติดอยู่ระหว่างโปรแกรมประยุกต์กับส่วนของฮาร์ดแวร์ โครงสร้างทั้ง 3 ส่วนได้แก่

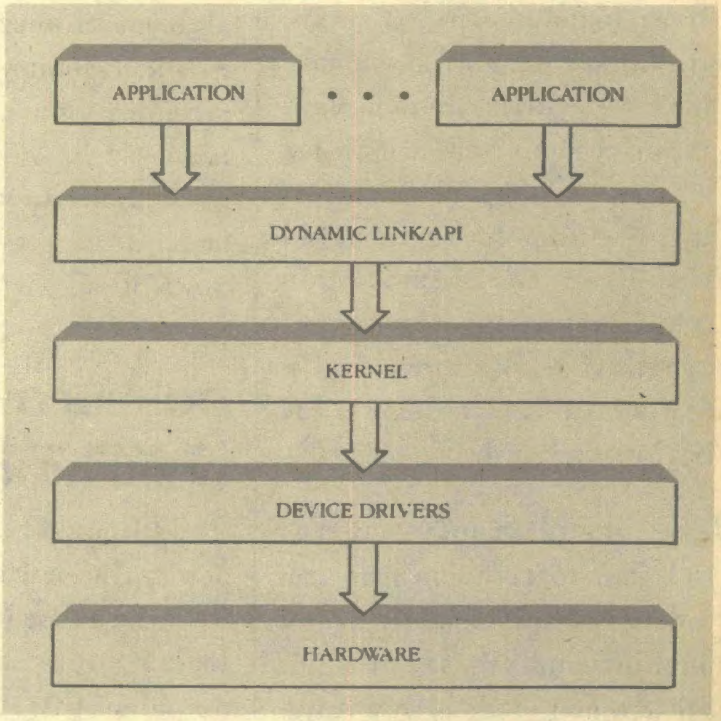
1. Subsystem
2. Kernel

และ 3. Device Drivers

ตัว Subsystem จะทำหน้าที่ในการเชื่อมต่อกับโปรแกรมประยุกต์หรือเรียกง่ายๆว่า API (Applications program interfaces) ในขณะเดียวกันก็เป็นส่วนสนับสนุนการใช้ Dynamic-Link-Libraries หรือ DLL ซึ่งจะอธิบายต่อไป

สำหรับส่วน Kernel หรือแก่นจะเป็นส่วนที่ทำหน้าที่บริการการทำงานพื้นฐาน หรือเป็นเสมือนศูนย์รวมของ OS/2 ก็ว่าได้ ตัว Kernel ประกอบด้วยส่วนย่อยๆอีกหลายส่วนคือ

- ระบบบริหารหน่วยความจำ (Memory Management)



รูปที่ 3
โครงสร้างของ OS/2 ซึ่งมีความยืดหยุ่นตัวต่อการปรับหรือแก้ไขโครงสร้างได้ง่ายกว่า MS-DOS

● ส่วนจัดลำดับงาน (Task Scheduling)

● ส่วนสื่อสารภายในระบบ (Interprocess Communication)

● ระบบอินพุตเอาต์พุตไฟล์ (File I/O)

● ส่วนบริการจังหวะเวลา (Timer Services)

● ส่วนบริหารสภาพแวดล้อม (Environment Management)

เราอาจจะพบว่าส่วนประกอบของ OS/2 ดูค่อนข้างจะสลับซับซ้อนและเข้าใจยากอยู่สักหน่อย จุดนี้ขอให้เราเพียงแต่ทำความเข้าใจกับองค์ประกอบหลักๆไว้ก่อน ส่วนรายละเอียดนั้นเป็นเรื่องปลีกย่อยภายหลัง

และสุดท้ายคือส่วนที่เป็นดีไวซ์-ไดรเวอร์ ซึ่งทำหน้าที่จัดการกับตัวฮาร์ดแวร์ ระบบดีไวซ์ไดรเวอร์นี้มีมาตั้งแต่เอ็มเอสดอสแล้ว แนวความคิดการใช้ดีไวซ์ไดรเวอร์นี้มีผลกระทบต่อแนวทางของเครื่องมือพัฒนาซอฟต์แวร์สมัยใหม่อยู่มาก กล่าวคือทำให้เราออกแบบระบบโดยแยกเป็นชิ้นๆที่ติดต่อกันโดยการส่งข่าวสารให้แก่กัน หรือเป็นโมดูลที่ทำหน้าที่กันอย่างอิสระ ทำให้การเปลี่ยนแปลงแก้ไขทำได้ง่าย กล่าวคือถ้ามีอุปกรณ์ฮาร์ดแวร์แบบใหม่เข้ามาก็เพียงแต่ใส่ไดรเวอร์ตัวใหม่เข้าไป ก็ทำให้โปรแกรมทำงานกับอุปกรณ์ฮาร์ดแวร์ใหม่ได้ทันที ซึ่งหลักการนี้จะตรงข้ามกับระบบมินิคอมพิวเตอร์ตรงที่การติดตั้งอุปกรณ์ฮาร์ดแวร์ใหม่บนมินิหรือปรับส่วนใดส่วนหนึ่งของตัวโอเอสจะต้องใช้เวลานานอย่างมาก

ในตอนนี้จะขออธิบายเรื่องของ DLL ที่กล่าวค้างไว้ในส่วน Subsystem ว่าคืออะไร และมีข้อดีอย่างไร ซึ่งก่อนอื่นขอให้ท่านผู้อ่านทำความเข้าใจกับคำว่าลิงก์ (Link) เสียก่อนว่าในที่นี้หมายถึงการเชื่อมต่อโมดูลของโปรแกรมหลายๆโมดูลเข้าด้วยกันกลายเป็นโปรแกรมใหญ่ที่ทำงานได้ โดยปกติแล้วการลิงก์ตามแบบฉบับของดอสเรียกว่าเป็นแบบ Static Link คือดึงเอาข้อมูล (หรือโปรแกรม) เข้ามารวมโดยตรงเลย แต่การลิงก์ตามแบบฉบับของ OS/2 กลับเป็นแบบ Dynamic Linking คือโมดูลไม่ได้ถูกโหลดมารวมกันจริง แต่เป็นเพียงคำสั่งเรียกใช้โมดูลเท่านั้นไม่ได้มีตัวโมดูลจริงๆเข้ามาอยู่รวมด้วย วิธีการแบบนี้จะเห็นได้ว่ามีข้อดีอย่างน้อยก็ 2 ข้อคือ

ประการแรก ขนาดไฟล์จะเล็กลง เนื่องจากไม่ได้บรรจุตัวโมดูลจริงๆไว้ แต่บรรจุเพียงคำสั่งเรียกใช้ซึ่งเมื่อมีการทำงานมาถึงจุดนี้จึงค่อยอ่านเอาโมดูลนั้นมาทำงาน

ประการที่สอง ก็คือโปรแกรมประยุกต์ต่างๆสามารถใช้โมดูลซึ่งมีเพียงก็อบปี้เดียวร่วมกันได้

เราจะเห็นได้แล้วว่าตัว Kernel เป็นแกนกลางของ OS/2 และ DLL เป็นส่วนของการเชื่อมต่อและขยาย ส่วนดีไวซ์ไดรเวอร์เป็นส่วนกำหนดอุปกรณ์ฮาร์ดแวร์ที่ใช้ร่วม ตัว OS/2 จึงมีลักษณะเป็นโมดูลที่สามารถขยายขีดความสามารถการทำงานหรือเพิ่มพูนการเชื่อมต่อโปรแกรมอื่นๆเข้ากับตัว OS/2 ได้ อย่างเช่น OS/2 รุ่น 1.1 ซึ่งมีการเพิ่มระบบกราฟิกอินเตอร์เฟสที่เรียกว่า Presentation Manager ตามแบบของไมโครซอฟต์วินโดว์

DOS กับ OS/2 เข้ากันได้เพียงใด

โปรแกรมที่ใช้งานอยู่บนเอ็มเอสดอสส่วนใหญ่จะนำมาใช้บน OS/2 ได้ ซึ่งส่วนใหญ่แล้วจะเป็นโปรแกรมที่เขียนโดยไม่อ้างอิงกับฮาร์ดแวร์นัก โปรแกรมเหล่านี้เมื่อนำมาใช้งานบน OS/2 จะทำงานอยู่ในสภาวะปกติ หรือ Real Mode เท่านั้น อย่าลืมว่าตัว OS/2 มีคุณสมบัติของโอเอสแบบมัลติทาสกิ้ง คือทำงานได้หลายงานพร้อมกัน งานที่กำลังทำอยู่และแสดงผลอยู่บนจอภาพด้วยเรียกว่าทำงานอยู่ข้างหน้าหรือ Foreground ส่วนงานที่ทำอยู่แต่ไม่แสดงให้เห็นบนจอภาพเรียกว่าทำอยู่ข้างหลังหรือ Background ผู้ใช้สามารถที่จะสลับสับเปลี่ยนงานใดๆมาเป็นโฟร์กราวนด์ หรือแบ็คกราวนด์ก็ได้ แต่มีข้อยกเว้นอยู่ว่าถ้านำเอาโปรแกรมที่ใช้งานอยู่บนเอ็มเอสดอสมารันบน OS/2 และให้โปรแกรมนั้นไปอยู่แบ็คกราวนด์แล้ว โปรแกรมนั้นจะไม่ทำงานเนื่องจาก OS/2 คาดเดาว่าโปรแกรมจากเอ็มเอสดอสนั้นอาจมีการเขียนข้อความลงบนจอภาพโดยตรง ซึ่งอาจทำให้ข้อความบนจอภาพของโฟร์กราวนด์เสียไปด้วย หรืออาจจะไปทำอย่างอื่นที่ขัดต่อการทำงานของโปรแกรมอื่นๆได้ ดังนั้นโดยสรุปแล้ว OS/2 จะทำงานให้เฉพาะแบ็คกราวน์ที่เขียนขึ้นเพื่อ OS/2 เท่านั้น

โปรแกรมประยุกต์... ผู้ใช้ยังต้องรอต่อไป

ถึงแม้ว่าการใช้ OS/2 จะทำให้ผู้ใช้สามารถนำเอาโปรแกรมประยุกต์ที่ใช้งานอยู่บนเอ็มเอสดอสมาใช้กับ OS/2 ได้ทันที แต่ก็ยังคงเป็นการใช้โปรแกรมที่มีขีดจำกัดของหน่วยความจำที่ 640 กิโลไบต์อยู่ดี และแท้ที่จริงแล้วเนื้อที่หน่วยความจำว่างสำหรับการรันโปรแกรมภายใต้ OS/2 จะลดลงไปเนื่องจากตัว OS/2 มีขนาดใหญ่กว่าดอสประมาณ 100 กิโลไบต์ (เทียบกับดอส 3.3 ขนาด 55 กิโลไบต์) ดังนั้นแล้วผู้ใช้ก็ยังคงต้องรอคอยโปรแกรมประยุกต์ตัวใหม่ที่จะนำมาใช้งานร่วมกับ OS/2 ได้อย่างจริงจัง